FEDERICO MASON

A Hierarchical Learning Approach to Enable Network Slicing in Future Telecommunication Systems

GIORNATA DI INCONTRO
BORSE DI STUDIO GARR
"ORIO CARLINI"
ROMA

UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

SUPERVISOR: PROF. ANDREA ZANELLA

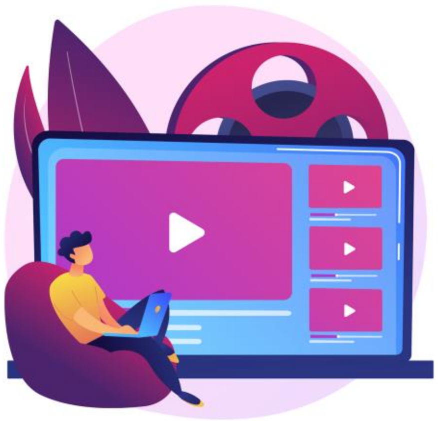
Roma, 25/11/2020

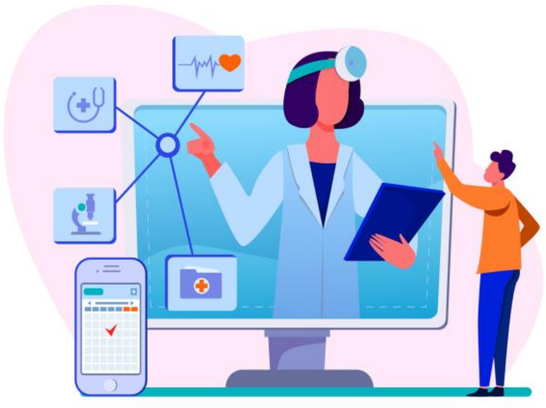
Borsisti Day 2020

INTRODUCTION

Future telecommunication systems will be characterized by **heterogeneous** applications with very **specific requirements**

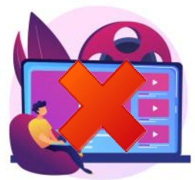
eMBB





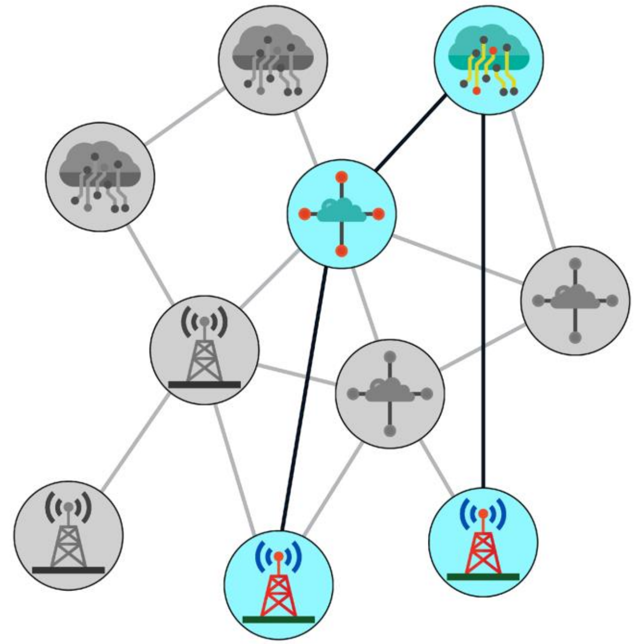
URLLC

INTRODUCTION



Current networks are too rigid to support such services

The **Network Slicing (NS)** paradigm makes it possible to define multiple **logical networks** over the same infrastructure



PROBLEM FORMULATION

In a NS scenario we need to dynamically **allocate system resources** among different network slices

- Very complex environment
- Difficult to be generalized

We attack the problem by designing a **Deep Reinforcement Learning (DRL)** architecture

SYSTEM MODEL

We consider a network where multiple **traffic flows** contend for the same communication and processing resources

Each flow is characterized by:

- Time-varying **requirements**
  - **Performance** function

That depend on the slice which the flow belongs to...

SYSTEM MODEL

There are two kinds of network elements

The **links** provide communication bandwidth and affect the flows' **throughput** and **delay**

The **nodes** provide computational and memory capacities to support the **network functions**

OUR GOAL

We want to maximize the system utility, which is given by

$$\Omega = \frac{1}{|\Phi|} \sum F_{\phi}$$

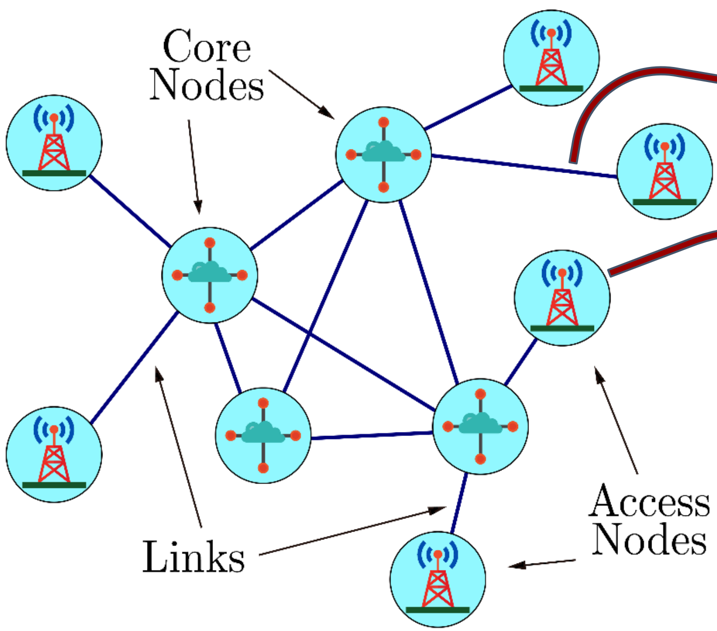
Total number of flows

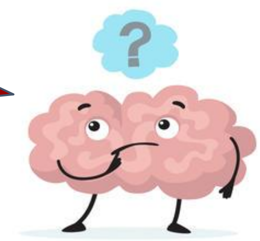
Sum over the flow set

Performance of flow ϕ

OUR APPROACH

We consider a distributed system, where each network element is associated with a **local controller**





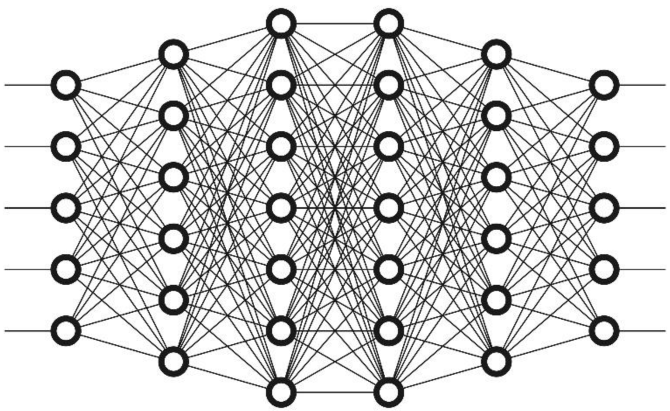
...manages link resources!

...manages node resources!

TRAINING

The number of controllers
depends on the network topology…

...but we train only two learning agents!



- We train agents by the **Advantage Actor Critic** algorithm
- Each agent is composed by two **Neural Networks**

TRAINING

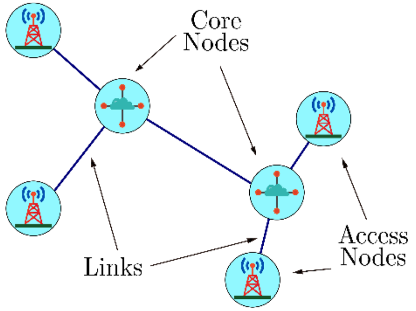
1) Observation: the agent see the status of a traffic flow crossing a network element

2) Action: the agent computes the amount of resources that the flow demands

3) Reward: the agent is rewarded according to the performance of the neighboring flows

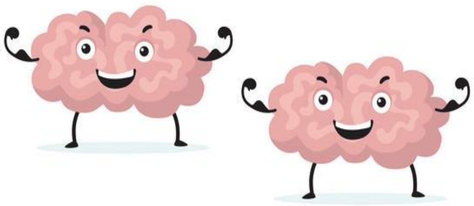
TRAINING

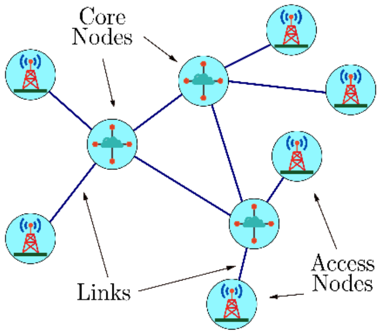
We consider two network topologies…



Dumbbell Network

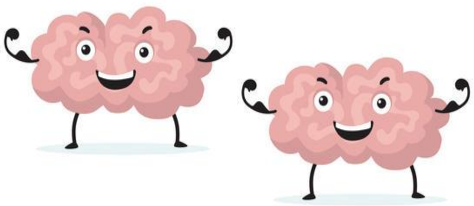
DRL-D





Triangle Network

DRL-T



BENCHMARK

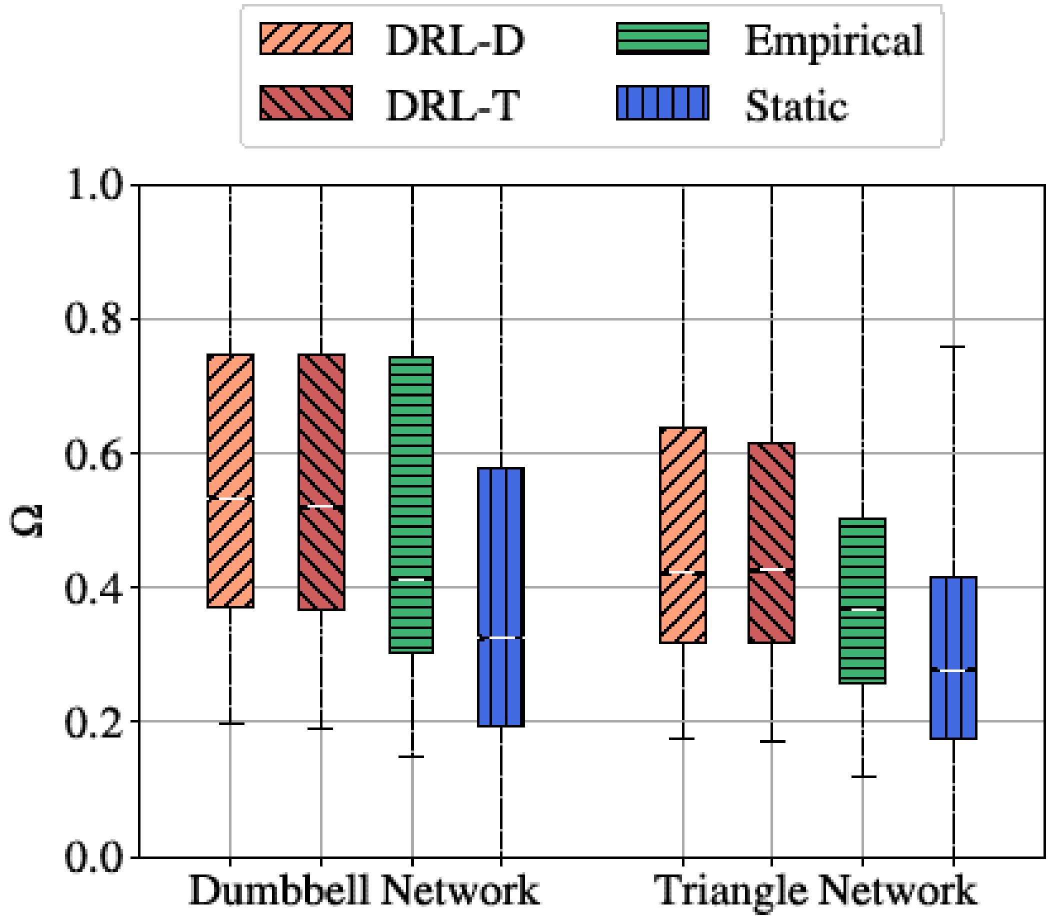
Static Strategy

- Centralized approach
- Consider the average flow demands
- It does not handle dynamic requirements

Empirical Strategy

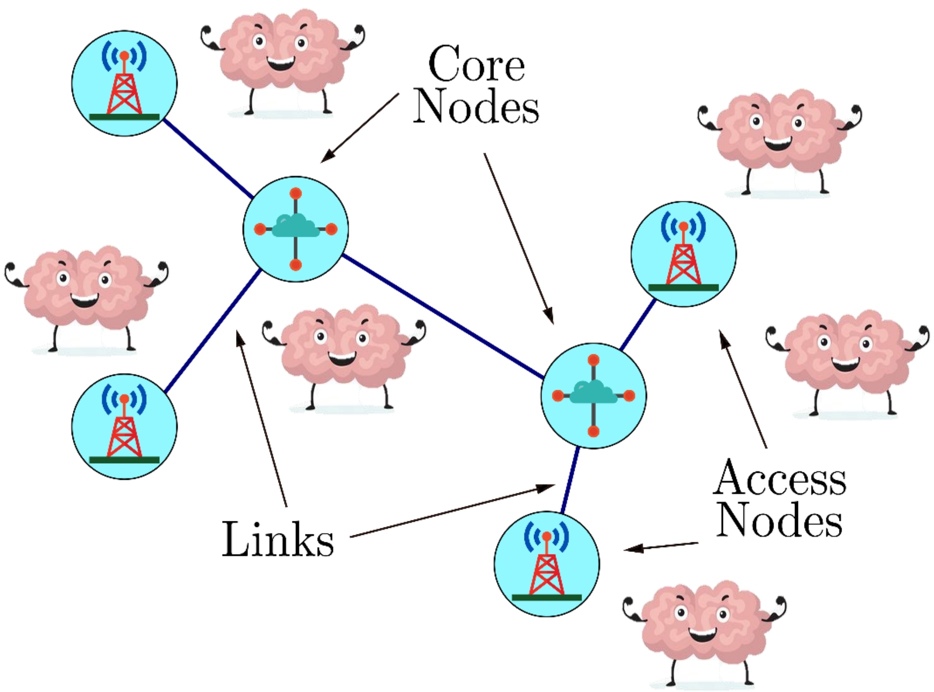
- Distributed approach
- Consider the instantaneous flow demands
- It does not handle slice diversity

RESULTS: DRL-D, DRL-T



TRANSFER LEARNING

Each agent is trained to operate in a specific network location

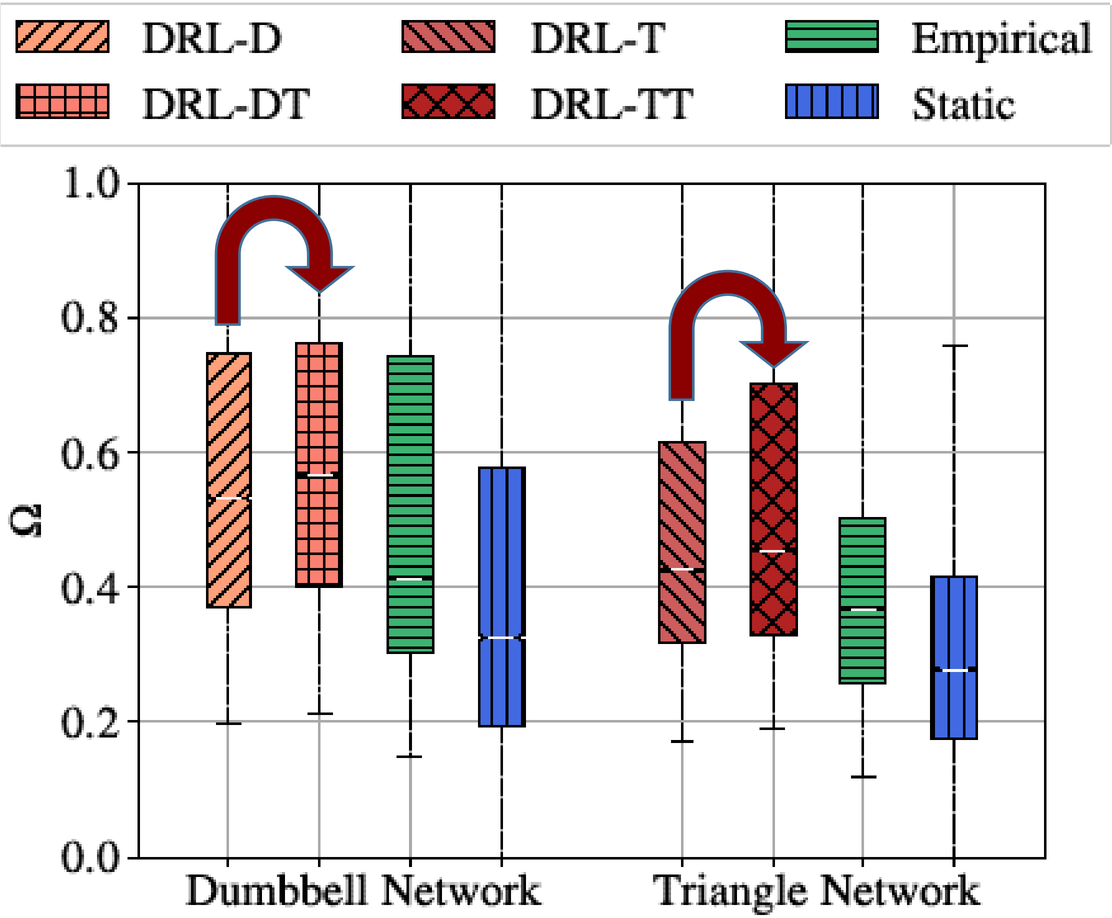


The number of agents increases!

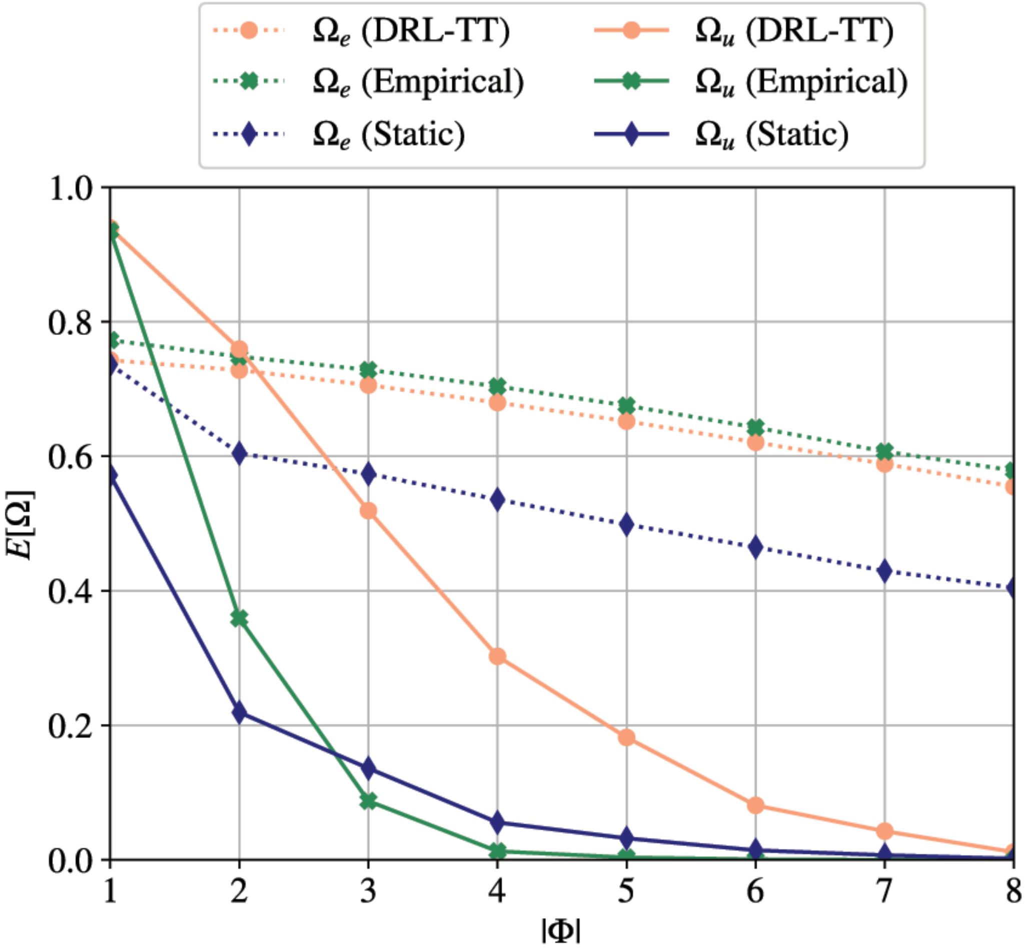
It is **more performant**…

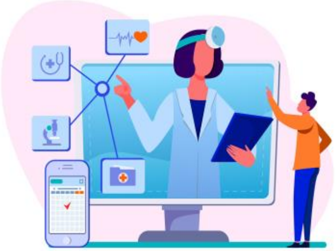
…but **less flexible!**

RESULTS: DRL-DT, DRL-TT



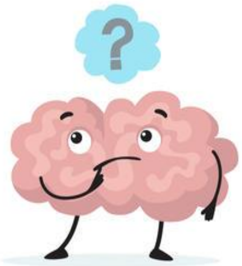
RESULTS: Utility vs Flow Number





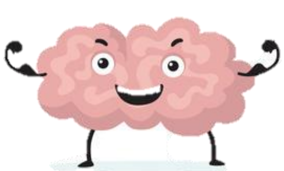
CONCLUSIONS

Our proposal **improve the orchestration** of network resources, especially when the system **complexity increases**



The same architecture can suit different network topologies **without additional training**

The **transfer learning** paradigm can be used to further improve the system utility



WHAT'S NEXT?

We are training additional learning agents to dynamically manage the routing of traffic flows

We want to implement our approach in **Public Safety Communication** scenario…

…to allow emergency operators to exploit **advanced technologies** in critical scenarios!

WHAT'S NEXT?

Consortium GARR
THE ITALIAN EDUCATION & RESEARCH NETWORK

Thank you for the attention!

Any question?

GIORNATA DI INCONTRO
BORSE DI STUDIO GARR
“ORIO CARLINI”
ROMA